



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/809,552	03/25/2004	Susann Marie Keohane	AUS920040035US1	7922
34533 7590 06/05/2007 INTERNATIONAL CORP (BLF) c/o BIGGERS & OHANIAN, LLP P.O. BOX 1469 AUSTIN, TX 78767-1469			EXAMINER WANG, BEN C	
			ART UNIT 2192	PAPER NUMBER
			MAIL DATE 06/05/2007	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/809,552	Applicant(s) KEOHANE ET AL.	
	Examiner Ben C. Wang	Art Unit 2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 25 March 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-20 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-20 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date <u>3/25/2004</u> | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. Claims 1-20 are pending in this application and presented for examination.

Specification Objections

2. The specification is objected to because the following informalities:
 - "include the 'Red Hat Package Manager' ("RPM")", cited in P.6 , Line 24, is a registered trademark
 - "entries in MSDOS, and entries in a Master File Table", cited in P.13 , Line 13, is a registered trademark
 - "from the executable file (204), a class to", cited in P. 12, Line 27, should be corrected as "from the executable file (304), a class to"
 - "And an exec() call give its newly created process", cited in P. 13, Line 27, should be corrected as "And an exec() call gives its newly created process"

Appropriate correction is required (See MPEP § 608.01(b))

Claim Rejections – 35 USC § 102(e)

The following is quotation of 35 U.S.C. 102(e) which form the basis for all obviousness rejections set forth in this office action:

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Art Unit: 2192

international application designated the United States and was published under Article 21(2) of such treaty in the English language.

3. Claims 1-20 are rejected under 35 U.S.C. 102(e) as being anticipated by Brenner et al. (Pat. No. US 6,859,926 B1) (hereinafter 'Brenner')

4. **As to claim 1**, Brenner discloses a method for assigning computational processes in a computer system to workload management classes (Abstract), the method comprising: installing on the computer system an executable file from a software installation package, wherein the software installation package includes a specification of workload management properties (Col. 5, Lines 39-46 – each process has associated attributes which may be used to perform a classification of the process into a defined class) for the executable file, including a definition of a workload management class (Figs. 5-6; Col. 6, Lines 32-49 - Class Assignment Rules – for a class to be defined, the class name, tier, resource shares and resource limits must be defined. Once a class has been defined, class assignment rules need to be created. The class assignment rules are used to assign processes to a class based on process attributes; Col. 6, Lines 49-60 – using these class assignment rules, processes are classified into various defined classes); executing a process in dependence upon the executable file (Fig. 11; Col. 12, Lines 50-57 – the operation starts with receiving processes for execution using a system resource; the assigned classes of the processes are determined; each class is guaranteed their minimum limit of the system resource; the system resource allocation is made based on relative shares of the classes and their class priority); and assigning the process to the

workload management class (Fig. 6; Col. 6, Lines 51-60 – classification rules are applied to processes; based on the attributes of the processes, e.g., user name, group name, fully qualified path, and the like, these processes meet certain requirements of various ones of the classification rules; as a result, the processes are classified into one of the predefined classes; as a result of the classification, these processes belong to classes which have an assigned tier value and number of shared which will be used to determine their access to system resources; Fig. 7D – a 'class assignment rules' GUI; Col. 7, Lines 24-33).

5. **As to claim 9**, Brenner discloses a system for assigning computational processes to workload management classes (Abstract), the system comprising: means for installing on a computer system an executable file from a software installation package, wherein the software installation package includes a specification of workload management properties (Col. 5, Lines 39-46 – each process has associated attributes which may be used to perform a classification of the process into a defined class) for the executable file, including a definition of a workload management class (Figs. 5-6; Col. 6, Lines 32-49 - Class Assignment Rules – for a class to be defined, the class name, tier, resource shares and resource limits must be defined. Once a class has been defined, class assignment rules need to be created. The class assignment rules are used to assign processes to a class based on process attributes; Col. 6, Lines 49-60 – using these class assignment rules, processes are classified into various defined classes); means for executing a process in dependence upon the executable file

Art Unit: 2192

(Fig. 11; Col. 12, Lines 50-57 – the operation starts with receiving processes for execution using a system resource; the assigned classes of the processes are determined; each class is guaranteed their minimum limit of the system resource; the system resource allocation is made based on relative shares of the classes and their class priority); and means for assigning the process to the workload management class (Fig. 6; Col. 6, Lines 51-60 – classification rules are applied to processes; based on the attributes of the processes, e.g., user name, group name, fully qualified path, and the like, these processes meet certain requirements of various ones of the classification rules; as a result, the processes are classified into one of the predefined classes; as a result of the classification, these processes belong to classes which have an assigned tier value and number of shared which will be used to determine their access to system resources; Fig. 7D – a 'class assignment rules' GUI; Col. 7, Lines 24-33).

6. **As to claim 15**, Brenner discloses a computer program product for assigning computational processes in a computer system to workload management classes (Abstract), the computer program product comprising: a recording medium (Fig. 2, elements 232 – hard disk, 209 – local memory); means, recorded on the recording medium, for installing on the computer system an executable file from a software installation package, wherein the software installation package includes a specification of workload management properties (Col. 5, Lines 39-46 – each process has associated attributes which may be used to perform a classification of the process into a defined class) for the executable

Art Unit: 2192

file, including a definition of a workload management class (Figs. 5-6; Col. 6, Lines 32-49 - Class Assignment Rules – for a class to be defined, the class name, tier, resource shares and resource limits must be defined. Once a class has been defined, class assignment rules need to be created. The class assignment rules are used to assign processes to a class based on process attributes; Col. 6, Lines 49-60 – using these class assignment rules, processes are classified into various defined classes); means, recorded on the recording medium, for executing a process in dependence upon the executable file (Fig. 11; Col. 12, Lines 50-57 – the operation starts with receiving processes for execution using a system resource; the assigned classes of the processes are determined; each class is guaranteed their minimum limit of the system resource; the system resource allocation is made based on relative shares of the classes and their class priority); and means, recorded on the recording medium, for assigning the process to the workload management class (Fig. 6; Col. 6, Lines 51-60 – classification rules are applied to processes; based on the attributes of the processes, e.g., user name, group name, fully qualified path, and the like, these processes meet certain requirements of various ones of the classification rules; as a result, the processes are classified into one of the predefined classes; as a result of the classification, these processes belong to classes which have an assigned tier value and number of shared which will be used to determine their access to system resources; Fig. 7D – a ‘class assignment rules’ GUI; Col. 7, Lines 24-33).

7. **As to claim 2** (incorporating the rejection in claim 1), **claim 10** (incorporating the rejection in claim 9), and **claim 16** (incorporating the rejection in claim 15), Brenner discloses the workload management class definition further comprises a class name, a priority ranking, and an inheritance attribute (Col. 6, Lines 33-34 – for a class to be defined, the class name, tier, resource shares and resource limits must be defined; Col. 9, Lines 59-65 – the tier priority adjustment is a value derived based on the relative priorities of the tiers).

8. **As to claim 3** (incorporating the rejection in claim 1), **claim 11** (incorporating the rejection in claim 9), and **claim 17** (incorporating the rejection in claim 15), Brenner discloses the specification of workload management properties further comprises minimum values and maximum values for CPU, memory, and disk I/O shares for the executable file (Col. 8, Line 59 through Col. 9, Line 4; Col. 7, Lines 1-5).

9. **As to claim 4** (incorporating the rejection in claim 1), Brenner discloses the method wherein installing an executable file further comprises: configuring the workload management class in dependence upon the workload management properties; and storing a class name of the workload management class in association with a pathname for the executable file (Fig. 5, elements of “class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3).

10. **As to claim 5** (incorporating the rejection in claim 1), Brenner discloses the method wherein installing an executable file further comprises storing a class name for the workload management class in association with a pathname for the executable file (Fig. 5, elements of “class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3).

11. **As to claim 6** (incorporating the rejection in claim 5), Brenner discloses the method wherein storing a class name for the workload management class in association with a pathname for the executable file further comprises storing the class name in the executable file (Fig. 5, elements of “class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3).

12. **As to claim 7** (incorporating the rejection in claim 5), Brenner discloses the method wherein storing a class name for the workload management class in association with a pathname for the executable file further comprises storing the class name in a data structure that represents the executable file in an operating system (Col. 5, Lines 39-45 – each process has associated attributes which may be used to perform a classification of the process into a defined class).

13. **As to claim 8** (incorporating the rejection in claim 5), Brenner discloses the method wherein assigning the process to the workload management class further comprises: identifying the workload management properties for the workload management class in dependence upon the pathname (Col. 6, Lines

Art Unit: 2192

38-42 – the process attributes utilized in this particular example for classification of processes are user name, group name and application path name); and configuring the workload management class in dependence upon the workload management properties (Fig. 5, elements of “class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3).

14. **As to claim 12** (incorporating the rejection in claim 9), Brenner discloses the system wherein means for installing the executable file further comprises: means for configuring the workload management class in dependence upon the workload management properties; and means for storing a class name of the workload management class in association with a pathname for the executable file (Fig. 5, elements of “class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3).

15. **As to claim 13** (incorporating the rejection in claim 9), Brenner discloses the system wherein means for installing an executable file further comprises means for storing a class name for the workload management class in association with a pathname for the executable file (Fig. 5, elements of “class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3).

16. **As to claim 14** (incorporating the rejection in claim 13), Brenner discloses the system wherein means for assigning the process to the workload management class further comprises: means for identifying the workload

Art Unit: 2192

management properties for the workload management class in dependence upon the pathname (Col. 6, Lines 38-42 – the process attributes utilized in this particular example for classification of processes are user name, group name and application path name); and means for configuring the workload management class in dependence upon the workload management properties (Fig. 5, elements of “class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3).

17. **As to claim 18** (incorporating the rejection in claim 15), Brenner discloses the computer program product wherein means, recorded on the recording medium, for installing the executable file further comprises: means, recorded on the recording medium, for configuring the workload management class in dependence upon the workload management properties; and means, recorded on the recording medium, for storing a class name of the workload management class in association with a pathname for the executable file (Fig. 5, elements of “class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3).

18. **As to claim 19** (incorporating the rejection in claim 15), Brenner discloses the computer program product wherein means, recorded on the recording medium, for installing an executable file further comprises means, recorded on the recording medium, for storing a class name for the workload management class in association with a pathname for the executable file (Fig. 5, elements of

Art Unit: 2192

“class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3)..

19. **As to claim 20** (incorporating the rejection in claim 19), Brenner discloses the computer program product wherein means, recorded on the recording medium, for assigning the process to the workload management class further comprises: means, recorded on the recording medium, for identifying the workload management properties for the workload management class in dependence upon the pathname (Col. 6, Lines 38-42 – the process attributes utilized in this particular example for classification of processes are user name, group name and application path name); and means, recorded on the recording medium, for configuring the workload management class in dependence upon the workload management properties (Fig. 5, elements of “class name” and “application”; Figs. 7A-7E; Col. 7, Line 1 through Col. 8, Line 3).

Conclusion

20. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

- Alford JR. et al., Graphical User Interface for Visualization of Sampled Data Compared to Entitled or Reference Levels (Pub. No. US 2003/0006988 A1)

Art Unit: 2192

- Brenner et al., Apparatus and Method of Integrating a Workload Manager with a System Task Scheduler (Pub. No. US 2003/0236815 A1)
- Brenner et al., System for Preserving Logical Partitions of Distributed Parallel Processing System after Rebooting by Mapping Nodes to their Respective Sub-Environments (Pat. No. 5,854,896)
- Brenner et al., Apparatus and a Method for Creating Isolated Sub0Environments Using Host Names and Aliases (Pat. No. 5,941,943)
- Brenner et al., Multiprocessor Load Balancing System for Prioritizing Threads and Assigning Threads into one of a Plurality of Run Queues Based on a Priority Band and a current Load of the Run Queue (Pat. No. US 7,080,379 B2)
- Davies et al., Workload Management Amongst Server Objects in a Client/Server Network with Distributed Objects (Pat. No. 6,003,083)

21. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ben C. Wang whose telephone number is 571-270-1240. The examiner can normally be reached on Monday - Friday, 8:00 a.m. - 5:00 p.m., EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on 571-272-3695. The fax

Art Unit: 2192

phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.



TUAN DAM
SUPERVISORY PATENT EXAMINER

BCW *fw*

May 22, 2007